Optimizing the Egyptian Coal Flotation Using an Artificial Neural Network

M. Farghaly[1], A. Serwa[2], M. Ahmed[3]

[1] Friedrich-Alexander University, Paul-Gordan str.3, 91052 Erlangen, Germany

[1] Al-Azhar University, Qena Branch, Qena, Egypt

[2]Assiut University, Assiut, Egypt

[1]m_galaly@yahoo.com; [2]ahmed_serwa@yahoo.com; [3]mamoah@yahoo.com.au

***Abstract-* A back forward Artificial Neural Network (ANN) was used to study the effect of flotation time, collector dosage, frother dosage, and impeller speed on flotation recovery and grade. The results of 13 flotation experiments conducted on Egyptian coal were used for training the network while another 28 experiments were used for validation. Simulation results showed that a one layer network with a [6 2] architecture was the one that gave the least standard error (SE). The values of this error were 6.66% and 0.37% with recovery and grade, respectively. The software used in this paper was designed to automatically select the network architecture according to the direction and position of the network error. Using this ANN to optimize the flotation process showed that the optimum flotation parameters were 207.6 seconds for the flotation time, 1865.6 g/t for the collector dosage, 828 g/t for the frother dosage, and 1754 rpm for the impeller speed. The results of optimization process showed that experiment number 135 gave the highest recovery and grade: 94.94% and 5.07% respectively.**

Keywords- Neural Network; Froth Flotation; Optimization; Coal Beneficiation

I. INTRODUCTION

Conventionally, coarse coal is processed through gravity separation systems and fine coal through flotation. Froth flotation is the well established process for fine coal cleaning. Flotation is based on exploiting differences in the surface properties of the organic material in coal and in mineral matter. The organic matter is hydrophobic, particularly in bituminous coals, and is not readily wetted as the mineral surfaces. Flotation is also dependent on coal rank, floatability increasing with increasing rank through the bituminous range [1].

Artificial Neural Network (ANN) is an attempt to simulate the basic functions of biological brain in order to perform complex functions by computer systems such as classifying objects, matching patterns, and predicting the future based on past information [2].

Artificial Neural Networks are computational systems that simulate the microstructures of a biological nervous system. The most basic components of ANNs are modeled after the structure of the brain.

The most basic element of the human brain is a specific type of cell, which provides us with the ability to remember, think, and apply previous experience to our every action. These cells are also known as neurons and each of these neurons can connect up to 200,000 other neurons. The power of the brain comes from the number of these basic components and multiple connections between them [3-7].

Basically, a biological neuron receives input from other sources, combines them in some way, performs a generally nonlinear operation on the result, and then outputs the final result. Biological neural networks are constructed in a three dimensional way from microscopic components [3-7]. Inspired by biological neurons, ANNs are composed of simple elements operating in parallel, that is, ANNs are the simple clustering of primitive artificial neurons. This clustering occurs by creating layers, which are then connected to one another.

As in nature, the network function is determined by the interconnection between the neurons. Each input to a neuron has a weight factor of the function that determines the strength of the interconnections and thus the contribution of that interconnection to the following neurons. ANNs can be trained to perform a particular function by adjusting the values of these weight factors between the neurons either from the information from outside the network or by the neurons themselves in response to the input. This is the key to the ability of ANNs to achieve learning and memory [8].

Artificial Neural Network (ANN) techniques have been the subject of extensive research in minerals processing and related subjects since long time [1-2, 9-13]. Cilek [11] used a feed forward back propagation neural network model to predict the effect of changing flotation variables on the number of cleaning and scavenging stages in a continuous flotation circuit. A back propagation feed forward neural network was also used to predict flotation recovery and grade from froth structure. Labidi et al. used an ANN to study the effect of air-flow rate, time, agitation speed and consistency on flotation kinetics during paper de-inking [13].

A multilayer feed forward ANN was used for optimizing a froth flotation process. The optimum values of flotation feed size, collector dosage and flotation cell impeller speed were determined by neural network [2].

In a recent work, the effect of feed means size, collector dosage, and impeller speed on flotation rate constant (K) was explored. ANN model were developed by training and validation of the network with the results of 35 flotation experiments [9].

Jorjani et al. [1] predicted the combustible value and combustible recovery of coal flotation concentrate by regression and Artificial Neural Network based on proximate and group macerals analysis.

An ANN based model predictive controller has been designed and implemented for controlling the froth depth (or interface level) in a laboratory scale pilot flotation column of diameter 10 cm and height 2.5 m by manipulating the tailings flow rate [10].

Massinaei and Doostmohammadi [12] discussed the development of empirical and Artificial Neural Network models which enable to predict bubble surface area flux in an industrial rougher column treating copper ores and validation of the models.

In this work, a back propagation neural network (BPNN) is used for optimizing a coal flotation process. The optimum values of flotation time, collector dosage, frother dosage and flotation cell impeller speed are determined by neural network. The ANN is trained and validated using data from flotation experiments done on samples of Egyptian coal using various combinations of the previously mentioned parameters.

II. TRAINING PROCESS FOR BACK-PROPAGATION ANNS

The multilayered neural network is the most widely applied neural network, which has been used in most researches so far. A back-propagation algorithm can be used to train these multilayer feed-forward networks with differentiable transfer functions. It performs function approximation, pattern association, and pattern classification. The term back propagation refers to the process by which derivatives of network error, with respect to network weights and biases, can be computed.

The training of ANNs by back propagation involves three stages [4]: (i) the feed forward of the input training pattern, (ii) the calculation and back propagation of the associated error, and (iii) the adjustment of the weights. This process can be used with a number of different optimization strategies.

A. Over-fitting problem

Multilayer feed-forward neural networks are believed to be able to represent any functional relationship between input and output if there are enough neurons in the hidden layers. However, too many neurons in the hidden layers may cause another problem, which is the so-called over fitting. The error on the training set is driven to a very small value because of the powerful ANN learning process, but when new data are presented to this network, the error will become large.

Clearly, when over fitting occurs, the network will not generalize well. The ideal method for improving network generalization is to use a network that is just large enough to provide an adequate fit. The larger the network, the more complex the functions that network can create. Therefore, if a small enough network is used, then it will not have enough power to over fit the data. Unfortunately, it is difficult to know beforehand how large a network should be for a specific application. There are two methods for improving generalization: regularization and early stopping [14].

Al-Thyabats [2] stated that network training and validation using a new set of data gave a large Mean Squared Error (MSE) with two different data sets. However, not all the predicted results show a large deviation from the experimental data. This may be due to noise in the experimental data or to over training of the ANN. This is one of the reasons why we searched about an automation determination of the best number of hidden layers and neurons. Other reasons like saving of the time, potential, and money were taken into consideration. In this research, few experiments were used for training process to avoid the problem of over training of the ANN.

III. BACK PROPAGATION NEURAL NETWORK (BPNN)

It is well known that Artificial Neural Network (ANN) architecture is based on the concepts of neurons, transfer functions and layers and their interconnections. Therefore, to design an ANN for a specific application, the designer should take into account the number of neurons, their arrangement in layers, the number of layers and the type of interconnections, the input weights and the type of transfer functions. The multilayer feed forward-back propagation network, which is one of the most common ANN architectures, was used in this work.

The number of neurons in the output layer was fixed (2 neurons) since flotation output was measured by two parameters: recovery and grade which have positive value between 0 and 100. The neural network architecture and the layer interconnections are shown in Fig. 1. The number of layers and neurons are usually determined automatically by the developed software.

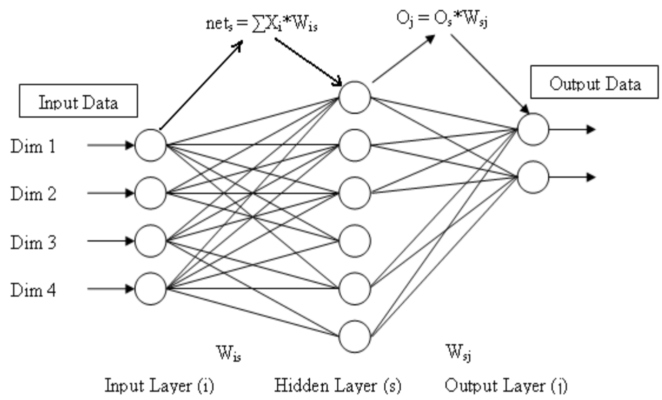

Fig. 1 Typical supervised BPNN architecture [after 8]

The BPNN algorithm is a generalized least squares algorithm that adjusts the connection weights between units to minimize the mean square error between the network output and the target output. Its architecture is MLP form. The target output is known from training data that it is the classes' values (experimental work results). The equations of this method are illustrated in [8, 15] and can be explained in the following sections.

Data entered to input unit are multiplied by the connection weights and is summed to derive the nets input to the unit in the hidden layer as shown in Fig. 1 and is given by:

$$net = \sum X_i \,.\, W_{is} \tag{1}$$

Where: Xi is a vector of inputs of an experiment or it is the vector of magnitude of ith input layer; Wis is matrix of the connection weights from ith input layer unit to sth hidden layer unit. Each unit in sth hidden layer computes a weighted sum of its inputs and passes the sum via an activation

function to the units in the jth output layer through weight vector Wsj.

There are a range of activation functions to transform the data from hidden layer unit to an output layer unit. These include pure linear, tangent hyperbolic, sigmoid functions etc. Although, the use of these functions may lead to difference in accuracy of classification [16], sigmoid function has been widely used and may be defined as:

$$O_s = \frac{1}{\left[1 + \exp^{-\lambda \,.\, nets}\right]} \tag{2}$$

Where Os is the output from the sth hidden layer unit and λ is a gain parameter, which controls the connection weights between the hidden layer unit and the output layer unit. Outputs from the hidden units are multiplied with the connection weights and are summed to produce the output of j^{th} unit in the output layer as:

$$O_j = O_s \,.\, W_{sj} \tag{3}$$

Where Oj is the network output for jth output unit (i.e. the land cover class) and Wsj is the weight of the connection between sth hidden layer unit and jth output layer unit.

An error function (E), determined from a sample of target (known) outputs and network outputs, is minimized iteratively. The process continues until E converges to some minimum value and the adjusted weights are obtained. E is given by:

$$E = \frac{1}{2} \,.\, \sum_{j=1}^{c} \left(T_j - O_j\right)^2 \tag{4}$$

Where Tj is the target output vector, O_j is the network output vector, and c is the number of classes. The target vector is determined from the known class allocations of the training pixels, which are coded in a binary form. The collection of known class allocations of all pixels will form the target vector.

After computing the error of the network, it is compared with the limiting error EL of the network. If E < EL, the network training is stopped, otherwise E is back propagated to the units in the hidden and the input layers. The connection weights are updated after several iterations. The number of iterations may vary from one dataset to the other, and is generally determined by trial and error. The process of back propagation and weight adjustment is explained in the following:

First, the error vector at each unit of the output layer is computed as:

$$E_j = O_j \,.\, (1 - O_j) \,.\, E \tag{5}$$

Then the error vector for each unit at the hidden layer is computed as:

$$E_s = O_s \,.\, (1 - O_s) \,.\, \sum_{j=1}^{j=c} O_j \,.\, E_s \tag{6}$$

Thereafter, the net error in connection weights between output layer and hidden layer is computed as:

$$E_{sj} = O_s \,.\, E_j \tag{7}$$

The error in connection weights between hidden layer and input layer can be determined as:

$$E_{is} = F_m \,.\, X_i \,.\, E_s \tag{8}$$

Where: Fm is the momentum factor which controls the momentum of the connections between the hidden layer unit and the input layer unit.

Once the error vectors are computed, the weight updating takes place for the next iteration. The weights between the output layer and hidden layer are updated as:

$$(W_{sj})_{new} = (W_{sj})_{old} + E_{sj} \tag{9}$$

and the weights between input layer and the hidden layer are updated as:

$$(W_{is})_{new} = (W_{is})_{old} + E_{is} \tag{10}$$

The gain parameter λ is also updated as:

$$\lambda_{new} = \lambda_{old} + L_R \,.\, E_s \tag{11}$$

where: L_R is the learning rate which controls the time of the learning process.

The next iteration starts with new set of weights and parameters and the process is repeated till the convergence is achieved, and the adjusted weights are obtained. At this stage, the network is assumed trained. The adjusted weights are then used to determine the outputs of the unknown pixels of the image using equations 1, 2 and 3. The network outputs are called activation levels. For hard classification, the pixel is allocated to the class with the highest activation level whereas for soft classification, these activation levels are scaled from 0 to 1 for a pixel to produce soft outputs [17]. The general procedure of BPNN algorithm may now be found in [8, 15].

A. Software Development

The Comparative software was developed to achieve the research objectives. The software is called Engineering Neural Network for Interpretation and Generalization of Meta Applications (ENIGMA). It was developed by Serwa using VB6 programming language. This software was designed to automatically select the network architecture according to the direction and position of the network error. The software starts with single hidden layer with single neuron and gradually increases the neurons or adding new hidden layer according to the type of network error. A schematic chart of BPNN Architecture of current software (ENIGMA) is revealed in Fig. 2. An illustration of the solving process of BPNN Architecture with the used software is shown in Fig. 3.

IV. NEURAL NETWORK SIMULATION

Different samples of Egyptian coal were used in froth flotation experiments. These samples were obtained from the main coal seam of Maghara mine, Sinia (Egypt). The head sample was stage crushed to minus 2 mm using a laboratory

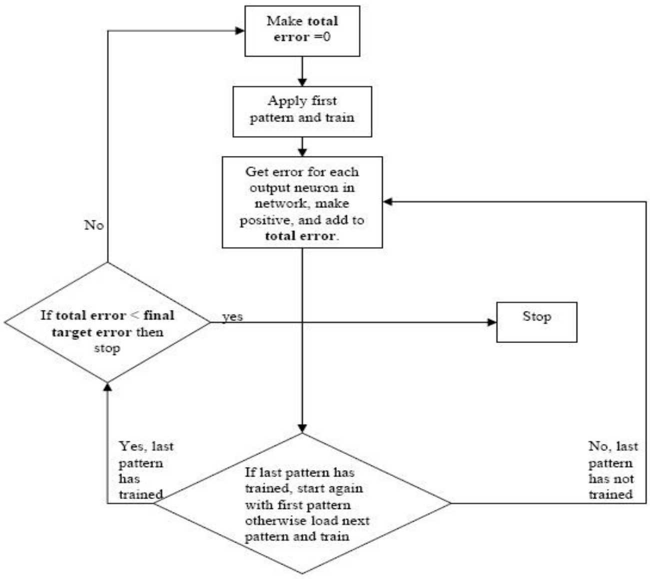

Fig. 2 A schematic chart of BPNN Architecture computer software

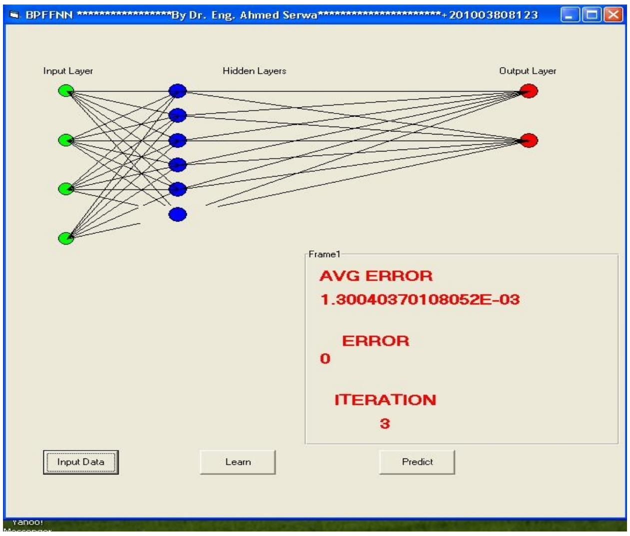

Fig. 3 The solving process of BPNN Architecture with the current software

Jaw crusher then a laboratory roll crusher. The crushed product was sieved to obtain the size below 0.5 mm.

All flotation tests were performed using commercially available kerosene as the collecting agent. It was changed between 740 and 1580 g/t. The frothing agent used was pine oil with different dosages between 225 and 675 g/t.

Flotation experiments were executed in three-liter Wemco flotation machine. All tests were carried out at pH = 7.0 and at a pulp density of 80 g/L. The impeller speed was changed between 950 and 1550 rpm and an airflow rate of 6 L/min was used. All tests were performed at room temperature. Concentrate grade and flotation mass recovery were used to measure flotation performance. The experiments were carried out at different combinations of flotation time (seconds), collector dosage (kg/TOF), frother dosage (kg/TOF), and flotation speed (rpm). The flotation time of experiments was ranged between 20 and 160 seconds. The collected products were filtered, dried, and weighed. The ash content of concentrate of each experiment was determined.

Table I shows the results of the flotation experiments. These results were randomly divided into two parts; the first part was used in network training while the second part was used for validation. The training and validation results are given in Tables II and III.

TABLE I
RESULTS FROM FLOTATION EXPERIMENTS

Exp. No.	Time, min.	Collector dosage, g/t	Frother dosage, g/t	Impeller speed, rpm	R_m, %	A_c, %
1	20	740	450	1250	33.89	4.41
2	40	740	450	1250	48.74	4.45
3	80	740	450	1250	69.75	4.86
4	160	740	450	1250	84.66	5.08
5	20	950	450	1250	39.11	4.58
6	40	950	450	1250	57.18	4.71
7	80	950	450	1250	76.76	4.94
8	160	950	450	1250	89.99	5.05
9	20	1160	450	1250	41.71	4.70
10	40	1160	450	1250	60.66	4.88
11	160	1160	450	1250	92.41	5.11
12	20	1370	450	1250	39.66	4.32
13	40	1370	450	1250	59.05	4.41
14	80	1370	450	1250	78.95	4.46
15	160	1370	450	1250	91.41	4.51
16	20	1580	450	1250	39.64	4.17
17	40	1580	450	1250	55.65	4.21
18	80	1580	450	1250	75.33	4.30
19	160	1580	450	1250	85.69	4.36
20	80	1055	225	1100	66.94	3.98
21	160	1055	225	1100	78.94	4.03
22	40	1055	338	1100	48.59	3.88
23	80	1055	338	1100	73.00	4.00
24	160	1055	338	1100	81.36	4.13
25	40	1055	450	1100	51.42	4.49
26	40	1055	563	1100	46.42	4.29
27	80	1055	563	1100	72.13	4.38
28	40	1055	675	1100	34.81	3.94
29	80	1055	675	1100	65.39	4.22
30	160	1055	675	1100	82.20	4.33
31	80	1265	338	950	48.11	3.96
32	40	1265	338	1100	47.46	3.77
33	80	1265	338	1100	69.32	3.82
34	160	1265	338	1100	82.13	3.97
35	40	1265	338	1250	62.94	3.87
36	160	1265	338	1250	89.65	4.15
37	40	1265	338	1400	66.72	4.77
38	160	1265	338	1400	93.84	4.86
39	40	1265	338	1550	63.03	4.27
40	80	1265	338	1550	82.70	4.35
41	160	1265	338	1550	93.47	4.67

The previously discussed Back Propagation Neural Network (BPNN) architecture was trained automatically and validated with these flotation results. An evaluation of these simulation results showed that network architecture with one hidden layer and 6 neurons gave the best training and validation results. Therefore, this arrangement of neural network architecture will be used to optimize flotation performance, as will be discussed in the next section.

TABLE II
TRAINING VALUES FOR DATA

Exp. No.	Time, min.	Collector dosage, g/t	Frother dosage, g/t	Impeller speed, rpm	R_m, %	A_c, %
1	20	740	450	1250	33.89	4.41
2	40	740	450	1250	48.74	4.45
5	20	950	450	1250	39.11	4.58
8	160	950	450	1250	89.99	5.05
9	20	1160	450	1250	41.71	4.70

11	160	1160	450	1250	92.41	5.11
12	20	1370	450	1250	39.66	4.32
13	40	1370	450	1250	59.05	4.41
15	160	1370	450	1250	91.41	4.51
25	40	1055	450	1100	51.42	4.49
27	80	1055	563	1100	72.13	4.38
30	160	1055	675	1100	82.20	4.33
41	160	1265	338	1550	93.47	4.67

TABLE IIII
TRAINING VALUES FOR DATA

Exp. No.	Time, min.	Collector dosage, g/t	Frother dosage, g/t	Impeller speed, rpm	R_m, %	A_c, %
3	80	740	450	1250	69.75	4.86
4	160	740	450	1250	84.66	5.08
6	40	950	450	1250	57.18	4.71
7	80	950	450	1250	76.76	4.94
10	40	1160	450	1250	60.66	4.88
14	80	1370	450	1250	78.95	4.46
16	20	1580	450	1250	39.64	4.17
17	40	1580	450	1250	55.65	4.21
18	80	1580	450	1250	75.33	4.30
19	160	1580	450	1250	85.69	4.36
20	80	1055	225	1100	66.94	3.98
21	160	1055	225	1100	78.94	4.03
22	40	1055	338	1100	48.59	3.88
23	80	1055	338	1100	73.00	4.00
24	160	1055	338	1100	81.36	4.13
26	40	1055	563	1100	46.42	4.29
28	40	1055	675	1100	34.81	3.94
29	80	1055	675	1100	65.39	4.22
31	80	1265	338	950	48.11	3.96
32	40	1265	338	1100	47.46	3.77
33	80	1265	338	1100	69.32	3.82
34	160	1265	338	1100	82.13	3.97
35	40	1265	338	1250	62.94	3.87
36	160	1265	338	1250	89.65	4.15
37	40	1265	338	1400	66.72	4.77
38	160	1265	338	1400	93.84	4.86
39	40	1265	338	1550	63.03	4.27
40	80	1265	338	1550	82.70	4.35

V. OPTIMISATION OF THE FLOTATION PARAMETERS

In previous sections some of the operating parameters for froth flotation were discussed. The aim was to explore the effect of these parameters on flotation recovery and concentrate grade of coal. The parameters and their ranges were: flotation time, 20-160 sec., collector dosage, 0.74–1.58 kg/TOF; frother dosage, 0.225–0.675 kg/TOF, and; impeller speed, 950–1550 rpm. In this section an ANN was used to predict flotation response at different combinations of these parameters. To simulate the process the range of each parameter was divided into 100 equal intervals. The simulation results are shown in Table 4. The results were plotted as the recovery-grade graph shown in Fig. 4. As shown in this Figure, experiment number 135 gave the highest recovery and grade: 94.94% and 5.07% respectively. The flotation parameters which gave these optimum values are 207.6 seconds flotation time, 1.866 kg/TOF collector dosage, 0.828 kg/TOF frother dosage, and 1754 rpm impeller speed.

TABLE IV
TRAINING VALUES FOR DATA

No.	Time, min.	Collector dosage, g/t	Frother dosage, g/t	Impeller speed, rpm	R_m, %	A_c, %
1	20.0	740.0	225.0	950	54.92	4.22
2	21.4	748.4	229.5	956	55.24	4.23
3	22.8	756.8	234.0	962	55.56	4.24
4	24.2	765.2	238.5	968	55.88	4.24
5	25.6	773.6	243.0	974	56.20	4.25
6	27.0	782.0	247.5	980	56.52	4.26
7	28.4	790.4	252.0	986	56.83	4.26
8	29.8	798.8	256.5	992	57.15	4.27
9	31.2	807.2	261.0	998	57.47	4.28
10	32.6	815.6	265.5	1004	57.79	4.28
11	34.0	824.0	270.0	1010	58.11	4.29
12	35.4	832.4	274.5	1016	58.42	4.29
13	36.8	840.8	279.0	1022	58.74	4.30
14	38.2	849.2	283.5	1028	59.05	4.31
15	39.6	857.6	288.0	1034	59.37	4.31
16	41.0	866.0	292.5	1040	59.68	4.32
17	42.4	874.4	297.0	1046	60.00	4.33
18	43.8	882.8	301.5	1052	60.31	4.33
19	45.2	891.2	306.0	1058	60.63	4.34
20	46.6	899.6	310.5	1064	60.94	4.35
21	48.0	908.0	315.0	1070	61.25	4.35
22	49.4	916.4	319.5	1076	61.57	4.36
23	50.8	924.8	324.0	1082	61.88	4.37
24	52.2	933.2	328.5	1088	62.19	4.37
25	53.6	941.6	333.0	1094	62.50	4.38
26	55.0	950.0	337.5	1100	62.81	4.39
27	56.4	958.4	342.0	1106	63.12	4.39
28	57.8	966.8	346.5	1112	63.43	4.40
29	59.2	975.2	351.0	1118	63.74	4.41
30	60.6	983.6	355.5	1124	64.05	4.41
31	62.0	992.0	360.0	1130	64.35	4.42
32	63.4	1000.4	364.5	1136	64.66	4.43
33	64.8	1008.8	369.0	1142	64.97	4.43
34	66.2	1017.2	373.5	1148	65.27	4.44
35	67.6	1025.6	378.0	1154	65.58	4.44
36	69.0	1034.0	382.5	1160	65.88	4.45
37	70.4	1042.4	387.0	1166	66.19	4.46
38	71.8	1050.8	391.5	1172	66.49	4.46
39	73.2	1059.2	396.0	1178	66.80	4.47
40	74.6	1067.6	400.5	1184	67.10	4.48
41	76.0	1076.0	405.0	1190	67.40	4.48
42	77.4	1084.4	409.5	1196	67.70	4.49
43	78.8	1092.8	414.0	1202	68.01	4.50
44	80.2	1101.2	418.5	1208	68.31	4.50
45	81.6	1109.6	423.0	1214	68.61	4.51
46	83.0	1118.0	427.5	1220	68.91	4.51
47	84.4	1126.4	432.0	1226	69.20	4.52
48	85.8	1134.8	436.5	1232	69.50	4.53
49	87.2	1143.2	441.0	1238	69.80	4.53
50	88.6	1151.6	445.5	1244	70.10	4.54
51	90.0	1160.0	450.0	1250	70.39	4.55

No.	Time, min.	Collector dosage, g/t	Frother dosage, g/t	Impeller speed, rpm	R_m, %	A_c, %
52	91.4	1168.4	454.5	1256	70.69	4.55
53	92.8	1176.8	459.0	1262	70.99	4.56
54	94.2	1185.2	463.5	1268	71.28	4.56
55	95.6	1193.6	468.0	1274	71.58	4.57
56	97.0	1202.0	472.5	1280	71.87	4.58
57	98.4	1210.4	477.0	1286	72.16	4.58
58	99.8	1218.8	481.5	1292	72.45	4.59
59	101.2	1227.2	486.0	1298	72.75	4.60
60	102.6	1235.6	490.5	1304	73.04	4.60
61	104.0	1244.0	495.0	1310	73.33	4.61
62	105.4	1252.4	499.5	1316	73.62	4.61
63	106.8	1260.8	504.0	1322	73.91	4.62
64	108.2	1269.2	508.5	1328	74.19	4.63
65	109.6	1277.6	513.0	1334	74.48	4.63
66	111.0	1286.0	517.5	1340	74.77	4.64
67	112.4	1294.4	522.0	1346	75.06	4.65
68	113.8	1302.8	526.5	1352	75.34	4.65
69	115.2	1311.2	531.0	1358	75.63	4.66
70	116.6	1319.6	535.5	1364	75.91	4.66
71	118.0	1328.0	540.0	1370	76.19	4.67
72	119.4	1336.4	544.5	1376	76.48	4.68
73	120.8	1344.8	549.0	1382	76.76	4.68
74	122.2	1353.2	553.5	1388	77.04	4.69
75	123.6	1361.6	558.0	1394	77.32	4.69
76	125.0	1370.0	562.5	1400	77.60	4.70
77	126.4	1378.4	567.0	1406	77.88	4.71
78	127.8	1386.8	571.5	1412	78.16	4.71
79	129.2	1395.2	576.0	1418	78.44	4.72
80	130.6	1403.6	580.5	1424	78.72	4.72
81	132.0	1412.0	585.0	1430	78.99	4.73
82	133.4	1420.4	589.5	1436	79.27	4.74
83	134.8	1428.8	594.0	1442	79.54	4.74
84	136.2	1437.2	598.5	1448	79.82	4.75
85	137.6	1445.6	603.0	1454	80.09	4.75
86	139.0	1454.0	607.5	1460	80.37	4.76
87	140.4	1462.4	612.0	1466	80.64	4.77
88	141.8	1470.8	616.5	1472	80.91	4.77
89	143.2	1479.2	621.0	1478	81.18	4.78
90	144.6	1487.6	625.5	1484	81.45	4.78
91	146.0	1496.0	630.0	1490	81.72	4.79
92	147.4	1504.4	634.5	1496	81.99	4.80
93	148.8	1512.8	639.0	1502	82.26	4.80
94	150.2	1521.2	643.5	1508	82.53	4.81
95	151.6	1529.6	648.0	1514	82.79	4.81
96	153.0	1538.0	652.5	1520	83.06	4.82
97	154.4	1546.4	657.0	1526	83.32	4.83
98	155.8	1554.8	661.5	1532	83.59	4.83
99	157.2	1563.2	666.0	1538	83.85	4.84
100	158.6	1571.6	670.5	1544	84.11	4.84
101	160.0	1580.0	675.0	1550	84.38	4.85

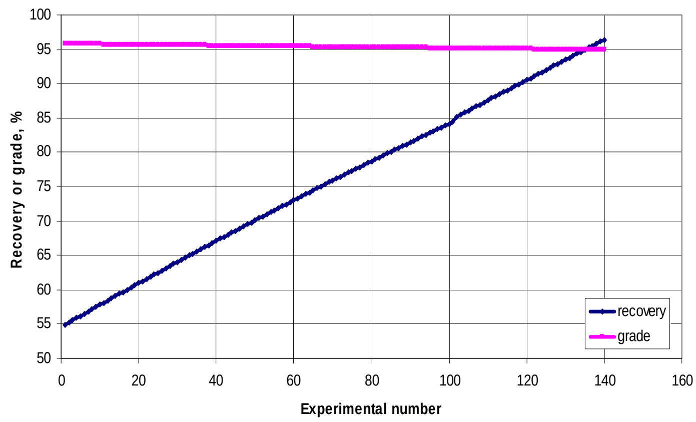

Fig. 4 The solving process of BPNN Architecture with the current software

VI. CONCLUSIONS

Samples of Egyptian coal were used to study the effects of flotation time, collector dosage, frother dosage, and impeller speed on flotation recovery and grade. Experiments were conducted in a Wemco flotation cell. The ranges of the operating parameters studied were 20-160 seconds for flotation time, 0.74–1.58 kg/TOF for collector dosage, 0.225–0.675 kg/TOF for frother dosage, and 950–1550 rpm for impeller speed. New software was tested for this work. This software selects automatically the network architecture according to the direction and position of the network error. The training results showed that an ANN with one hidden layer [6 2] was the one that gave the smallest MSE. To use an ANN for optimizing a flotation process, the range of each of the parameters was divided into 100 equal steps and simulated using the ANN net. Plotting the results on a recovery-grade graph showed that the highest values for recovery and grade were 94.94% and 5.07%, respectively. The optimum flotation parameters to give these results were 207.6 seconds flotation time, 1865.6 g/t collector dosage, 828 g/t frother dosage, and 1754 rpm impeller speed.

REFERENCES

[1] Jorjani, E., Asadollahi Poorali, H., Sam, A., Chehreh Chelgani, S., Mesroghli, Sh., and Shayestehfar, M.R., "Prediction of coal response to froth flotation based on coal analysis using regression and artificial neural network", Minerals Engineering, Vol. 22, pp. 970–976, 2009.

[2] AL-THYABAT, S., "On the optimization of froth flotation by the use of an artificial neural network", J China Univ Mining & Technol, Vol. 18, pp. 418–426, 2008.

[3] Aleksander, I., Morton, H. (1990), An Introduction to Neural Computing, 1st ed., Chapman & Hall, London, [ManualRequest][Infotrieve],2009,www.emeraldinsight.com/Insight/html/Output/Published/EmeraldFullTextArticle/Pdf/0291020903_ref.html

[4] Fausett, L., Fundamentals of Neural Networks, Englewood Cliffs, NJ: Prentice Hall Fausett, 1994

[5] Haykin, S., Neural networks: A comprehensive foundation, 2nd ed. Prentice-Hall, 1998.

[6] Bishop, C., Neural Networks for Pattern Recognition. Oxford UP, 1995.

[7] Swingler, K., Applying Neural Networks: A Practical Guide, London: Academic Press. Part 2, 1996.

[8] Serwa A., "Automatic Extraction of Topographic Features from Digital Images"., PhD Thesis, Al-Azhar University, Cairo, Egypt, 2009.

[9] AL-THYABAT, S., "Investigating the effect of some operating parameters on phosphate flotation kinetics by neural network", Advanced Powder Technology, Vol. 20, pp. 355–360, 2009.

[10] Mohanty, S., "Artificial neural network based system identification and model predictive control of a flotation column", Journal of Process Control, Vol. 19 pp. 991–999, 2009.

[11] Cilek, E.C., "Application of neural networks to predict locked cycle flotation test results", Minerals Engineering, Vol. 15, pp. 1095–1104, 2002.

[12] Massinaei, M. and Doostmohammadi, R., "Modeling of bubble surface area flux in an industrial rougher column using artificial neural network and statistical techniques", Minerals Engineering, Vol. 23, pp. 83–90, 2010.

[13] Labidi, J., Pe`lach, M.A., Turon, X., Mutje´, P., "Predicting flotation efficiency using neural networks", Chemical Engineering and Processing, Vol. 46, pp. 314–322, 2007.

[14] Demuth H. and M. Beale, Neural network toolbox. User's Guide. (Version 4),2000.http://www.mathworks.com/access/helpdesk/help/pdf_doc/nnet/nnet.pdf

[15] Ibrahim, M.A., "Evaluation of Soft Classifiers for Remote Sensing Data". Ph.D. Thesis. Department of Civil Engineering, Indian Institute of Technology, Reokree, India, 2004

[16] Ozkan, C. and Erbek, F. S., "The Comparison of Activation Functions for Multispectral Landsat TM Image Classification", Photogrammetric Engineering and Remote Sensing, 69, 1225-1234, 2003.

[17] Foody, G. M., 1996b. "Relating the Land-Cover Composition of Mixed Pixels to Artificial Neural Network Classification Output", Photogrammetric Engineering and Remote Sensing 62, pp. 491-499, 1996.